Homework 5

CMSY-217, Spring 2011

The source code for this assignment must be submitted electronically using the CE6 course website prior to the start of class on Thursday, April 28.

The Social Security Administration provides a webpage (http://www.ssa.gov/OACT/babynames) with two interactive applications that allow you to display the most popular baby names for a selected year and track the popularity of a selected baby name over several years. The data are based on Social Security Card applications for years 1880 through 2009.

Top 10 Names for 2009

Rank	Male name	Female name
1	Jacob	Isabella
2	Ethan	Emma
3	Michael	Olivia
4	Alexander	Sophia
5	William	Ava
6	Joshua	Emily
7	Daniel	Madison
8	Jayden	Abigail
9	Noah	Chloe
10	Anthony	Mia

Popularity of the female name Chloe

Year of birth	Rank
2009	9
2008	10
2007	16
2006	18
2005	19
2004	23
2003	24
2002	25
2001	30
2000	38
1999	63

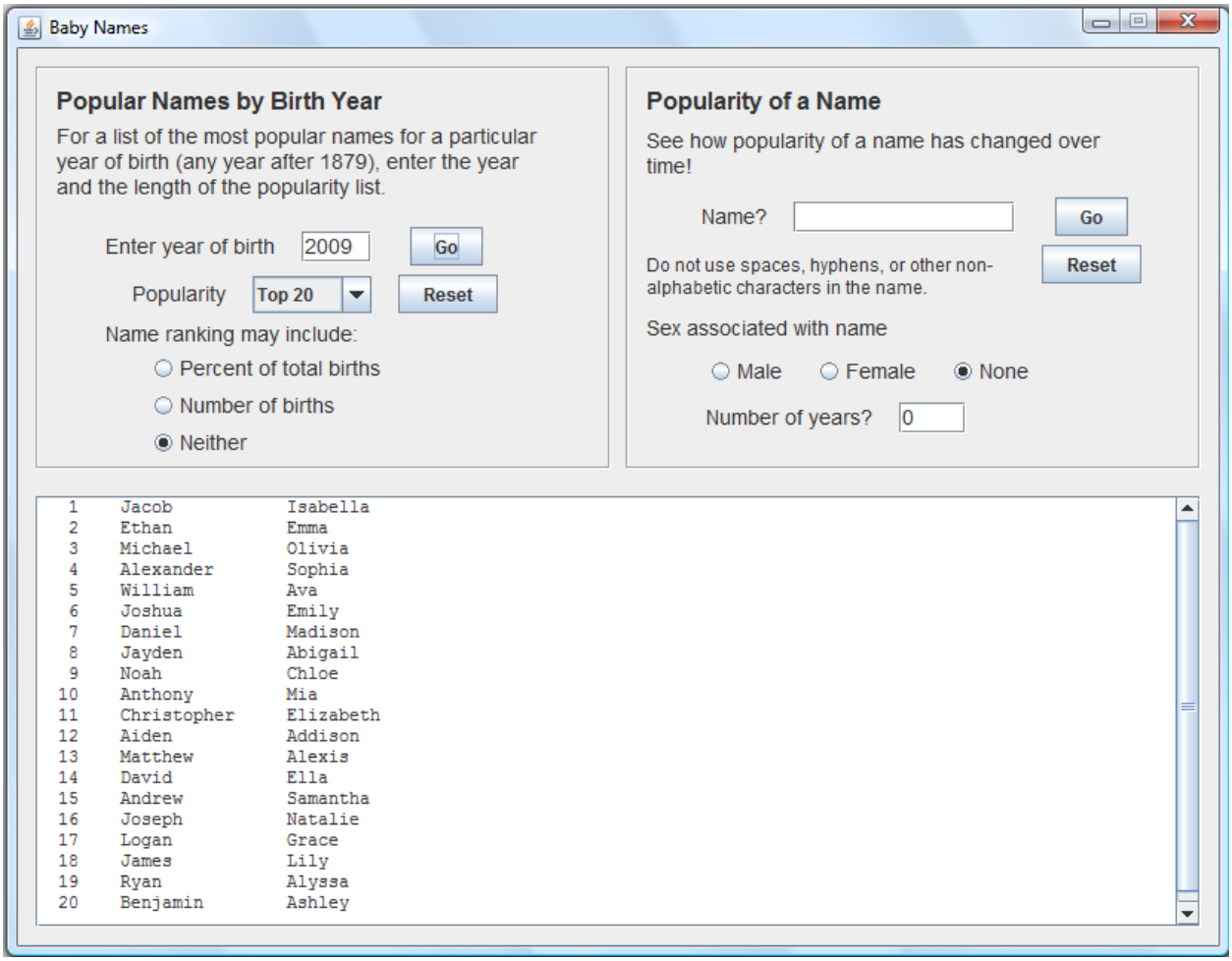
The data have been placed in a Java DB (Apache Derby) database named `babynames` with a table for each year of birth that contains columns for name, sex, and number of births. Each table is named with the letters `YOB` followed by the four-digit year for which the data was collected. For example, the table `YOB2009` contains the data for year 2009. Note that each name entry in these tables begins with a capital letter and is follwed by all lowercase letters.

```
ij> DESCRIBE YOB2009;
COLUMN_NAME         |TYPE_NAME
-----------------------------
NAME                |VARCHAR
SEX                 |CHAR
NUMBER              |INTEGER
```

There is also a table called `TOTALBIRTHS` which contains the total number of births, by gender, for each year.

```
ij> DESCRIBE TOTALBIRTHS;
COLUMN_NAME         |TYPE_NAME
-----------------------------
BIRTHYEAR           |INTEGER
MALE                |INTEGER
FEMALE              |INTEGER
```

The `BabyNames` class contains a Swing application that provides you with a graphical user interface (GUI) similar in appearance to the HTML forms on the Social Security website. In addition, the event-handling code has been written so that when the user clicks the `Go` button - a `BabyNamesQuery` object is created, the input parameters are passed to the `getList` or `getRank` method, and a results `String` is returned which is displayed in the `JTextArea` at the bottom of the GUI. The following figure shows the `BabyNames` application running with the results of a `getList` method call displayed.



1. Write the `getList` method in the `BabyNamesQuery` class using JDBC to provide the same functionality as the **Popular Names by Birth Year** application on the Social Security website.

2. Write the `getRank` method in the `BabyNamesQuery` class using JDBC to provide the same functionality as the **Popularity of a Name** application on the Social Security website.

The following SQL statements are examples of the String query objects that you could pass to the executeQuery method of the Statement interface to return a ResultSet object. Figure 28.23 from the textbook would be a good starting point for the necessary Java code.

```
SELECT NAME
  FROM YOB2009
  WHERE SEX='M'
  ORDER BY NUMBER DESC, NAME ASC
  FETCH FIRST 20 ROWS ONLY

SELECT NAME, NUMBER
  FROM YOB2009
  WHERE SEX='M'
  ORDER BY NUMBER DESC, NAME ASC
  FETCH FIRST 20 ROWS ONLY

SELECT MALE
  FROM TOTALBIRTHS
  WHERE BIRTHYEAR=2009

SELECT NUMBER
  FROM YOB2009
  WHERE NAME='Chloe' AND SEX='F'

SELECT COUNT (NAME)
  FROM YOB2009
  WHERE SEX='F' AND NUMBER > 11785
     OR SEX='F' AND NUMBER = 11785 AND NAME<='Chloe'
```